PATENT INFORMATION FORM

Oscode(s):	Inventor(s):	ISC	PHNL021237	
RO1242	Hekstra A.	R8F	PS:03-01	
	Baggen S.	R8E	PS:	
	Coene Wim	R8N	PS:	
			ID: 609244	
Priority Date(s):	Countries:		Client ref:	
27-11-2002	EP		COS	
			PE	WAC
			OUDH	7

Short title: Stripe wise trellis-based bit detection for 2D optical storage

Use within Philips: opical storage 4th G (Two DOS project)

Remarks:

Abstract and drawing:

The invention relates to trellis-based symbol detection for detecting symbols of a channel data stream recorded on a record carrier. Typically, it applies when symbol values are recorded along a two-dimensional (2D) channel tube forming a multi-track wide spiral comprising a set of adjacent symbol tracks aligned with respect to each other in the radial direction. The invention enables to do a kind of per "small number of" tracks equalization, so-called wise equalization to decode the 2D information encoded on the broad spiral. The simplest embodiment of the invention uses a 2-track Viterbi bit detector.

In accordance with the invention, the 2-track Viterbi bit detection comprises the following steps. A first estimation (equalization) is made of the bits in the top two tracks. Then, using the fact that the first track is adjacent to the guard space between the successive revolutions of the broad spiral, side information (e.g. zeroes) is derived in the computation of the hypothetical channel outputs that label the branches of the Viterbi trellis. This significantly increases the reliability with which the tracks adjacent to the guard space can be estimated

Fig. 6 illustrates an embodiment using thirty (3 iterations needed for 11 rows) 2-track Viterbi bit-detector machines. Once the first 2-track Viterbi bit-detector has output its first estimates of the first (and second) track, with a certain fixed, quite small delay, a second 2-track Viterbi bit-detector can start working on the 2nd (and 3rd) track, while using the output of the first Viterbi machine for the first track as side information etc. This way, 10 machines are needed per iteration, for three iterations it amounts to 30 Viterbi machines.

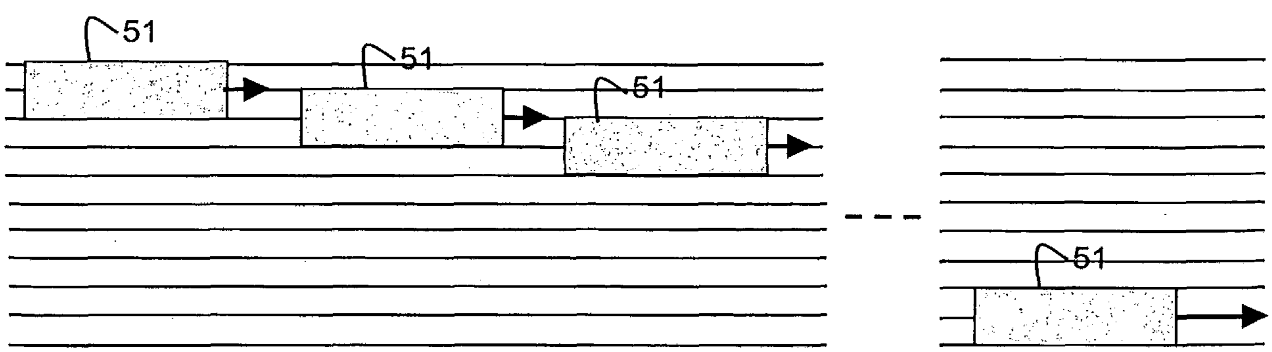


FIG. 6

DEMANDE DE BREVET EUROPEEN –1er Dépôt
EUROPEAN PATENT APPLICATION –1st Filing

TITRE OFFICIEL OFFICIAL TITLE	: ITERATIVE STRIPEWISE TRELLIS-BASED SYMBOL DETECTION METHOD AND DEVICE FOR MULTI-DIMENSIONAL RECORDING SYSTEMS.
INVENTEUR(S) – INVENTOR(S)	: HEKSTRA Andries P., COENE Willem, BAGGEN Stan
N° ENREGISTREMENT (DEPOT) OFFICIAL FILING NUMBER	: 02 292937.6
DATE DE DEPOT-FILING DATE	: November 27, 2002
DEMANDEUR-APPLICANT	: KONINKLIJKE PHILIPS ELECTRONICS N.V.
MANDATAIRE-PATENT ATTORNEY	: VAN OUDHEUSDEN Laure et al Philips IP&S-FR (SPID) 156, Bd Haussmann - 75008 PARIS FRANCE Tél : 33 (0) 1 40 76 80 30 - FAX 33 (0) 1 45 61 05 36 Liste Spéciale INPI 422.5/S008
Référence Philips IP&S-FR (SPID) Correspondant à - corresponding to	: PHNL021237 EPp LVO/LVO/OUDH - JR : ID 609244

The invention relates to trellis-based symbol detection for detecting symbols of a channel data stream recorded on a record carrier. Typically, it applies when symbol values are recorded along a two-dimensional (2D) channel tube forming a multi-track wide spiral comprising a set of adjacent symbol tracks aligned with respect to each other in the radial direction. The invention enables to do a kind of per "small number of" tracks equalization, so-called wise equalization to decode the 2D information encoded on the broad spiral.

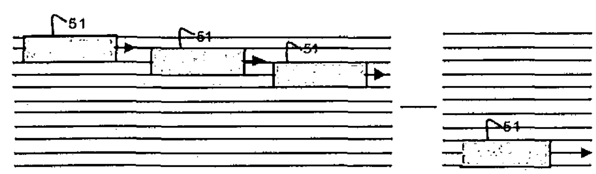


FIG. 6

Fig. 6

ITERATIVE STRIPEWISE TRELLIS-BASED SYMBOL DETECTION METHOD AND DEVICE FOR MULTI-DIMENSIONAL RECORDING SYSTEMS

FIELD OF THE INVENTION

The invention relates to a trellis-based symbol detection method for detecting symbols of a channel data stream recorded on a record carrier.

The invention applies to digital recording systems, such as magnetic recording and optical recording systems. It is particularly advantageous for two-dimensional optical recording, which is one of the potential technologies for the next generations of optical recording.

BACKGROUND ART

Current state-of-the-art optical disc systems are based on one-dimensional (1D) Optical Recording. A single laser beam is directed at a single track of information, which forms a continuous spiral on the disc, spiraling outwards to the outer edge of the disc. The single spiral contains a single (or one dimensional, 1D) track of bits. The single track consists of sequences of very small pit-marks or pits and the spaces between them, which are called land-marks or lands. The laser light is diffracted at the pit structures of the track. The reflected light is detected on a photo-detector Integrated Circuit (IC), and a single high-frequency signal is generated, which is used as the waveform from which bit-decisions are derived.

A new route for the 4th generation of optical recording technology that will succeed "Blue Ray Disc" also called "DVR" already succeeding DVD (Digital Video Disc) technology is based on two-dimensional (2D) binary optical recording. 2D recording means that e.g. 10 tracks are recorded in parallel on the disc without guard space in between. Then, the 10 tracks together form one big spiral. The format of a disc for 2D optical recording (called in short a "2D disc") is based on that broad spiral, in which the information is recorded in the form of 2D features. The information is written as a honeycomb structure and is encoded with a 2D channel code, which facilitates bit-detection. The disc shall be read out with an array of e.g. 10 (or more) optical spots, which are sampled in time, in order to obtain a two dimensional array of samples in the player. Parallel read out is realized using a single laser beam, which passes through a grating, which produces the array of laser spots. The array of spots scans the full width of the broad spiral. The light from each laser spot is reflected by the 2D pattern on the

disc, and is detected on a photo-detector IC, which generates a number of high-frequency signal waveforms. The set of signal waveforms is used as the input of the 2D signal processing. The motivation behind 2D recording is that much less disc space is wasted as guard space, so that the recording capacity of the disc can be increased. Although 2D recording is first studied for optical recording, similarly, magnetic recording can also be made two-dimensional.

One of the new aspects of such recording techniques is that they require two dimensional signal processing. In particular, one optical spot must be considered as a device which takes a plane of "pits"/"lands" (or "marks" and "non-marks") as input and produces a corresponding output. The optical spot transfer function has the characteristics of a 2D low pass filter, whose shape can be approximated by a cone. Apart from its linear transfer characteristics, the 2D optical channel also has non-linear contributions. The radius of the cone corresponds to the cutoff frequency, determined by the numerical aperture of the lens, and the wavelength of the light. This filtering characteristic causes 2D Inter Symbol Interference (ISI) in the player. It is the task of a bit-detector to annihilate (most of) this ISI (which can be both linear and non-linear). An optimal way to implement a bit-detector is to use a Viterbi algorithm. A Viterbi bit-detector does not amplify the noise. If soft decision output, i.e. reliability information about the bits, is required, a dual-Viterbi i.e (Max-)(Log-)MAP, or MAP, or SOVA (Soft Output Viterbi) algorithm can be used.

One of the difficulties of designing a bit-detector for the 2D case, is that a straightforward Viterbi bit-detector would need as its "state", one or more columns of "old" track bits because of the memory of the ISI. If e.g. 10 tracks are recorded in parallel in the 2D broad spiral, and e.g. two old bits per track is needed for a proper description of the state because of the tangential extent (along-the-tracks) of the 2D impulse response, this results in a state of 2 x 10 = 20 bits. Thus, the number of states in the Viterbi (or MAP, (Max-)(Log-)MAP, MAP, SOVA, etc.) algorithm becomes 2^{20}, which is completely impractical. This requires a different approach, which may be slightly sub optimal, but has a significantly reduced complexity.

SUMMARY OF THE INVENTION

It is an object of the invention to provide a symbol-detector that is much less complex than a brute force Viterbi or MAP bit-detector for multi-dimensional recording systems.

In accordance with the invention, it is provided a symbol detection method for detecting the symbol values of a channel data stream recorded along an N-dimensional

channel tube, N being at least 2, on a record carrier of a set of symbol rows, one-dimensionally evolving along a first direction and being aligned with each other along at least a second of N-1 other directions, said first direction together with said N-1 other directions constituting an N-dimensional lattice of symbol positions, the method comprising iterative stripe-by-stripe application of a trellis-based symbol detection step, wherein a stripe is a subset of at least one neighboring row. Using a stripe wise application of a trellis-based symbol detection method enables to reduce the number of states in the trellis, which makes the implementation less complex.

In accordance with particular embodiments of the invention, the method comprises application of a Viterbi, (Max-)(Log-)MAP or SOVA algorithms or their derivatives. Depending on the algorithm, better performance or reduced complexity can be obtained.

In accordance with a preferred embodiment of the invention, an iteration of said stripe wise trellis-based symbol detection step comprises:
- estimating symbol values in a first subset of rows, denoted current subset, using a trellis algorithm, side information derived from at least one row adjacent to said current subset being used in the estimation of symbol values,
- shifting to a second subset of rows.

Taking into account side information derived from adjacent rows enables to account for the Inter Symbol Interference (ISI) and thus improve the reliability of the symbol detection.

The invention advantageously applies when said symbol values are recorded along a two-dimensional channel tube forming a multi-row wide spiral comprising a set of adjacent symbol rows aligned with respect to each other in the radial direction. In this case, a symbol-free guard space row is provided between successive revolutions of the multi-row wide spiral. Then, in accordance with a preferred embodiment of the invention, side information is derived from said guard space row, the content of which is completely known, since the guard band is empty (all bits are zero). Therefore, in most embodiments, the first subset of rows processed is advantageously adjacent to the guard space row.

Generally, the present invention is applicable to a multi-dimensional code, where the channel words of the channel data stream may evolve in more than one direction, as is the case for a card-based system, i.e. where the channel data stream is recorded on a record carrier along a multi-dimensional channel tube with dimension at least two. Therein, the first direction along which the symbol rows evolve is preferably common to all symbol rows of the channel tube. The first direction constitutes together

with the N-1 other directions, along which the symbol rows are aligned with each other, an N–dimensional space and an N-dimensional lattice of symbol positions. The channel tube comprises at least two symbol rows of channel symbols evolving along said first direction, and the collection of all said channel tubes fill the whole N–dimensional space.

However, it is preferred to apply the invention to a channel data stream which comprises a one-dimensionally evolving symbol sequence or which comprises a broad spiral that comprises several consecutive channel strips stacked one upon the other, with each channel strip consisting of at least one symbol row one-dimensionally evolving along a first direction and aligned with each other along a second direction in the case there are at least two rows in a strip, preferably oblique or even orthogonal to said first direction, said two directions constituting a two-dimensional lattice of symbol positions. Preferred embodiments of such a lattice are a 2D lattice of a square or a hexagonal type.

The invention also relates to:

- a symbol detector,
- a method of reproduction of a user data stream,
- a reproduction device for reproduction of a user data stream,
- a receiver for receiving codeword symbols from a channel medium,
- a recording medium,
- a system comprising a source for delivering codeword symbols through a channel medium and a receiver for receiving said codeword symbols from the channel medium,
- a computer program product for carrying out the methods mentioned above,
- a signal for carrying the computer program and
- making available for downloading the computer program.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention and additional features, which may be optionally used to implement the invention, are apparent from and will be elucidated with reference to the drawings described hereinafter and wherein:

- Fig. 1 is a conceptual diagram for illustrating an example of a system in accordance with the invention,
- Fig.2 is a conceptual diagram for illustrating the principle of strip-based 2D coding that can be implemented for coding data on a recording medium in accordance with the invention,

- Fig. 3A and 3B are diagrammatic top-views of 1D and 2D spirals, respectively, for illustrating both 1D and 2D coding concepts,
- Fig. 4 is a diagram for illustrating a typical Signal-Pattern in accordance with a preferred embodiment of the invention,
- Fig. 5 is a conceptual diagram for illustrating an example of a symbol detecting method in accordance with a fist embodiment of the invention,
- Fig. 6 is a conceptual diagram for illustrating an example of a symbol detecting method in accordance with a second embodiment of the invention,
- Fig. 7 shows curves of the Bit Error Rate versus Peak Signal to Noise Ratio performance of various bit-detectors.

DETAILED DESCRIPTION OF THE DRAWINGS

The following remark relate to reference signs. Same block labels in all figures usually indicate same functional entities. Also, it is to be noted that when reference is made to a recording medium, it applies to both read-only and rewritable media.

Fig. 1 shows an example of a system in accordance with the invention. It comprises a source 1 for transmitting input symbols DI through a channel data stream 2, and a receiver 3 for receiving the symbols from the channel data stream and for delivering output symbols DO. In a typical data recording system wherein which the invention can be implemented, the source 1 comprises transmission means 10, 20, 30, 40, for transmitting input symbols DI from a user data stream, the channel data stream 2 is a record carrier 50 for recording the transmitted symbols of the user data stream, and the receiver 3 comprises reception means 60, 70, 80, 90, for retrieving output symbols DO from said record carrier. In particular, the receiver comprises a symbol detector 70 for detecting the symbol values retrieved from said record carrier.

Typical coding and signal processing elements of such data recording system will now be described with respect to Fig. 1. The cycle of user data from input DI to output DO can include interleaving 10, error-correction-code (ECC) and modulation encoding 20, 30, signal preprocessing 40, data recording on the recording medium 50, signal post-processing 60, binary detection 70, and decoding 80, 90 of the modulation code, and of the interleaved ECC. The ECC encoder 20 adds redundancy to the data in order to provide protection against errors from various noise sources. The ECC-encoded data are then passed on to a modulation encoder 30, which adapts the data to the channel, i.e. it manipulates the data into a form less likely to be corrupted by channel errors and more easily detected at the channel output. The modulated data are then input to a recording device, e.g. a spatial light modulator or the like, and recorded in

the recording medium 50. On the retrieving side, the reading device (e.g. photo-detector device or charge-coupled device (CCD)) returns pseudo-analog data values, which must be transformed, back into digital data (one bit per pixel for binary modulation schemes). The first step in this process is a post-processing step 60, called equalization, which attempts to undo distortions created in the recording process, possibly in the pseudo-analog domain. Then the array of (pseudo-analog) values is converted to an array of binary digital data via a bit detector 70. The array of digital data is then passed first to the modulation decoder 80, which performs the inverse operation to modulation encoding, and then to an ECC decoder 90.

In CD (Compact Disc), DVD and DVR, the physical coding format is based on a one-dimensional single spiral. In 2D optical recording, the concept of a broad spiral is introduced, which consists in a two-dimensional area built up from a number of adjacent rows of bits, or tracks, stacked one upon the other in a coherent fashion on a common underlying lattice of bits.

In the European patent application EP 01203878.2 the 2D constrained coding on hexagonal lattices in terms of nearest-neighbor clusters of channel bits is described. Therein, it has been focused mainly on the constraints with their advantages in terms of more robust transmission over the channel, but not on the actual construction of such 2D codes. The latter topic is addressed in the European patent application 02076665.5, i.e. the implementation and construction of such a 2D code is described therein. By way of example, a certain 2D hexagonal code shall be illustrated in the following. However, it should be noted that the general idea of the invention and all measures can be applied generally to any 2D code, in particular any 2D hexagonal or square lattice code. Finally, the general idea can also be applied to multi-dimensional codes, possibly with isotropic constraints, characterized by a one-dimensional evolution of the code. In the following a 2D hexagonal code shall be considered.

Fig. 2 illustrates the principle of strip-based 2D coding used in 2D optical recording that can be implemented for coding data on a recording medium such as e.g. an optical disc. The code evolves along a one-dimensional first direction (tangential i.e. parallel to the tracks of the disc). A 2D strip consists of a number of 1D rows or tracks, stacked upon each other in a second direction (e.g. radial) substantially orthogonal to the first (tangential) direction. The broad spiral consists of a coherent stacking of strips one upon the other. Between successive revolutions of the broad spiral a guard band of, for instance, one row high may be located. The bits on the 2D hexagonal lattice can be identified in terms of bit clusters. A hexagonal cluster may consist of a bit at a

central lattice site, surrounded by six nearest neighbors at the neighboring lattice sites. Also larger more complex cluster can be devised.

Fig. 3A and 3B are diagrammatic top-views of 1D and 2D spirals, respectively, for illustrating both 1D and 2D coding concepts, the latter being based on a hexagonal lattice of bits. Channel bits in the broad spiral are indicated as cells of a honeycomb structure.

Fig. 4 shows a typical "Signal-Pattern". The signal-levels for 2D recording on hexagonal lattices are identified by a plot of amplitude values for the complete set of all hexagonal clusters possible. Use is further made of the isotropic assumption, that is, the channel impulse response is assumed to be circularly symmetric. This implies that, in order to characterize a 7-bit cluster, it only matters to identify the central bit, and the number of "1"-bits (or "0"-bits) among the nearest-neighbor bits (0, 1, ..., 6 out of the 6 neighbors can be a "1"-bit). A "0"-bit is a land-bit in our notation. Assuming a broad-spiral consisting of 11 parallel bit rows, with a guard band of 1 (empty) bit row between successive broad spirals, the situation of Fig. 4 corresponds to a density increase with a factor of 1.7 compared to traditional 1D optical recording (as used in e.g. in the Blue-ray Disc (BD) format (using a blue laser diode).

A general threshold level to be applied for all signal levels of the HF-signal has been drawn, which will lead to optimum bit-error rate for threshold detection. However, it is obvious that the signal levels of the two bottom clusters with the central bit equal to a "0"-bit, and the signal levels of the two top clusters with central bit equal to a "1"-bit, are at the wrong side of the threshold level, that is, threshold detection would lead obviously to erroneously detected (central) bits in these cases. These clusters are part of the so-called error-zone (or erasure-zone), as depicted in Fig. 4. The probability of occurrence of these clusters (with almost 100% probability of error) amounts to (1+6)/64, which is about 11%, leading to a BER of about 1.1 10-1. Obviously, this BER is by far too high.

The gist of the invention is to do a kind of per "small number of" tracks bit-detection, so-called stripe-by-stripe or stripe wise bit-detection to decode the 2D information encoded on the broad spiral. The simplest embodiment of the invention uses a 2-track Viterbi bit detector. Examples described hereafter with respect to the following figures relate to this simple embodiment, which is not restrictive. Using any n-track (n>0) trellis-based symbol detector will fall within the scope of the invention. Nevertheless, using stripes of 1 track high gives worse BER (Bit Error Rate) versus SNR (Signal to Noise Ratio) performance. Fig. 5 illustrates the detecting method in accordance with this simple embodiment of the invention, which uses a 2-track Viterbi

bit detector 51. The principle of the one-dimensional Viterbi bit detection, which comprises only a single row of bits, is well known in the state of the art for one-dimensional modulation and coding. It is for instance described in Chapter7, particularly in paragraphs 7.1, 7.2, 7.3 and 7.5 "Viterbi Detection" by Jan Bergmans, "Digital Baseband Transmission and Recording", Kluwer Academic Publishers, 1996 [1].

In accordance with the invention, the 2-track Viterbi bit detection comprises the following steps. A first estimation (equalization) is made of the bits in the top two tracks. Then, using the fact that the first track is adjacent to the guard space between the successive revolutions of the broad spiral, side information (e.g. zeroes) is derived in the computation of the hypothetical channel outputs that label the branches of the Viterbi trellis. This significantly increases the reliability with which the tracks adjacent to the guard space can be estimated Bit-detection errors might occur in the first two tracks due to the fact that the bits in the 3rd, 4th, etc... track are unknown and are not modeled correctly in this decoding attempt. E.g. they all can be set to only zeroes, ones, random, alternating 0-1, or a "special" value which lies in between 0 and 1 and which indicates a channel output value in between a corresponding channel input value of 0 or 1. Another alternative for these bits outside of the current two-track stripe is to use preliminary bit-decisions coming from a very simple threshold detector, which is described in relation to Fig. 4. Since the 2nd track was closest to these unknown tracks it is deemed the most unreliable.

Next, the 2-track Viterbi bit-detector is shifted one track down, and is operated again over the 2nd and 3rd track. Now, the hard-decision bits of the 1st track and 4-th track serve as side information in the computation of the hypothetical channel outputs that label the branches of the trellis etc. until all tracks have been detected. During the detection of the 2nd and 3rd track, the estimated bits for the 2nd and 3rd track are overwritten. Like this, each time, the track closest to the guard space remains while the other is overwritten. In general, the reliability of the estimated bits closest to the guard space will be the highest.

When the procedure is restarted again at the first track for a second iteration of the same procedure, a first estimate for the bits of the 3rd (and 4th) track, that are adjacent to the top two tracks, is already available for the computation of the hypothetical channel outputs that label the Viterbi trellis to be used as side information at tracks above and tracks below the current stripe of tracks to be detected. Thus, during this second pass, the number of bits in error will generally decrease significantly due to better side information obtained from the previous iteration of the stripe-wise detection over the complete broad spiral, or a part thereof.

Numerical experiments showed that for some typical recording densities after three passes of the 2-track bit-detector through all tracks, the bit error rate saturates, i.e. no longer decreases significantly. It is to be noted, that each time the 2-track Viterbi bit-detector is lowered by one, one track of equalized bits is produced that is the result of that pass; the lower of the two tracks is overwritten when we lower the 2-track bit-detector by one track. In this sense, the new proposed bit-detector is a recursive detector in the radial direction of the disc, by succession of the 2-track Viterbi detector in the tangential direction each time at a 1-track shifted position. Since the 2-track Viterbi detector is in its turn a recursive detector, it results in a twofold recursive bit-detector both along tangential (parallel to the tracks) direction and along the radial (or substantially radial) direction of the broad spiral. Using sliding window or next iteration initialization techniques can allow for parallelization along the track direction.

It should be observed that in the example illustrated in Fig. 5, when two bits of memory are needed per track for the bit-detection, it results for the trellis in only two bits per track x two tracks (= four bits) per state. Thus, the number of states in one trellis section is only $2^4=16$. This is a notable reduction with respect to the 2^{20} states, which were needed without the invention.

Simulations for typical densities show that the degradation in bit error rate due to the stripe wise approach of the invention is quite acceptable, e.g. increases by a factor of 2. Bit error rates (BER) are generally measured on a logarithmic scale, thus a doubling of the bit error rate, generally, is quite acceptable. In total, e.g. three passes or iterations are required to obtain a low BER.

In the sequel, several alternative embodiments of the inventions are mentioned, which are not restrictive.

Fig. 6 illustrates an embodiment using thirty (3 iterations needed for 11 rows) 2-track Viterbi bit-detector machines.

Once the first 2-track Viterbi bit-detector has output its first estimates of the first (and second) track, with a certain fixed, quite small delay, a second 2-track Viterbi bit-detector can start working on the 2nd (and 3rd) track, while using the output of the first Viterbi machine for the first track as side information etc. This way, 10 machines are needed per iteration, for three iterations it amounts to 30 Viterbi machines. In case a train of 10 2-track Viterbi bit detectors is used, for three repeats of the iterative procedure ("iterations of iterations"), the hard bits obtained during the stripe wise processing one repeat earlier on row i+1 is used for row i-1, i during the second repeat. The hard bits of adjacent tracks to the stripe are needed to compute the channel outputs above the stripe for all the branches in the trellis based processing.

In an implementation, this can be realized by means of an array of hard-decision bits with as many rows as there are in one big spiral, which are filled with stripe wise processing per stripe (e.g. one row, preferable 2 rows). Already during the first time going down the rows, the hard bit decision for the bits in row i-1 is needed when the stripe contains row i (and e.g. also row i+1). If the trellis based processing of a stripe produces soft information, this must be quantized to get hard bit decisions to feed these decisions into the processing of the next stripe. Using soft info for the adjacent tracks may be better but it would lead to a marked increase in complexity.

Alternatively, it is also possible to make the Viterbi machines operate on three, or more tracks at the same time, and then shift the machine down, 1, 2 (or more) tracks, etc. In the following a plethora of extensions and generalizations of the invention are considered.

The (Max-)(Log-)MAP algorithm, or any trellis-decoding algorithm, could be used instead of a Viterbi algorithm. Descriptions of such algorithms can be found e.g. in "Turbo Codes: Principles and Applications", by Branka Vucetic and Jinhong Yuan, Kluwer Academic Publishers, Boston/Dordrecht/London 2000, especially in Chapter 5, notably in paragraph 5.7 for the MAP Algorithm, in 5.8 for the Max-Log-MAP Algorithm and in 5.9 for the Log-MAP Algorithm. In case of the (Max-)(Log-)MAP algorithm, a soft-decision-input, soft-decision-output (SISO) sub optimal bit-detector would be obtained, whereas the Viterbi algorithm itself produces a hard-decision output bit estimate. Such a SISO bit-detector can advantageously be combined with error correcting codes that are SISO-decodable, such as Low Density Parity Check (LDPC) codes, or Turbo codes for example.

Because the bit-detection for 2D optical recording requires a very high data rate processing, sliding window Viterbi, (Max-)(Log-)MAP, SOVA, etc. trellis-decoding algorithms can advantageously be used in the bit-detection method instead of a pure Viterbi algorithm, in order to speed-up the decoding process. It is well known that splitting the trellis into overlapping windows, so-called the sliding window technique, can speed up trellis-decoding algorithms. Then, a separate hardware unit can be provided to process each of these sliding windows.

Also next iteration initialization can be done instead of sliding window or (Max-)(Log-)MAP algorithms. Next iteration initialization (NNI) was used in Turbo codes. It is referred to C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, A. Glavieux, "Iterative corrrection of intersymbol interference: Turbo equalization," European Transactions on Telecommunications, Vol. 6, pp. 507-511, Sept. 1995 [2] for an introduction to this technique. The (Max-)(Log-)MAP algorithms require the

computation of metrics in the forward direction of the trellis and in the backward direction. With NNI, the metrics in the backward direction are recomputed per sliding window, from certain trellis stages, for which these metrics are copied from an earlier iteration. This allows avoiding the overlap of sliding windows for initialization purposes. If fully parallel processing of all windows is intended, copying of the metrics at the start of the window is required for the forward metric computations and at the end of the windows for the backward computations.

Alternatively, a soft decision output algorithm can be used only during iterations, which are followed by decoding of an error correction code. This refers to the situation in which the current bit-detector outputs its bit estimates to a soft-input decoding algorithm of some error correcting code, e.g. an LDPC or Turbo code. In general, soft-output bit-detection requires more computation that hard-decision output. If e.g. three iterations of the bit-detection scheme is used as described above, then, during the first two iterations, e.g. the Viterbi algorithm, which produces hard-decision bit estimates, can be used and during the last iteration, e.g. (Max-)(Log-)MAP, which requires twice as much computations as the Viterbi algorithm, can be used to produce soft decision outputs for the error correcting code decoder.

Also, available *a priori* information, that comes e.g. from a soft decision decodable error correcting code ("Turbo equalization") can also be used and fed into the algorithm. In a preferred implementation, *a priori* probabilities of 0.5 for all bits are used for all iterations. Similarly, in the case of Turbo equalization described in [2], *a priori* probabilities can be used in the stripe detector equal to the output extrinsic information of the soft decision error correcting (e.g. LDPC) code decoder in the previous Turbo loop. That is, soft output information is not generated inside the bit-detector that is fed back as *a priori* information for the bit-detector. Although this is possible, it gives no significant performance increase, or even gives a performance decrease.

To dampen the influence of the *a priori* information, a fudge factor can also be used. In case of Turbo equalization described in the cited reference with convolutional codes, e.g. Turbo codes or LDPC or other soft decodable error correcting codes, the *a priori* information fed into the bit-detector may be reduced in reliability by e.g. using a fudge factor (multiplier smaller than one) on the log-likelihood. In case one would want to feed back soft information within the bit-detector, one can also use fudge factors, if desired.

Side information tracks can also be included in the state of the trellis algorithm. In a preferred implementation, the side information tracks are represented as hard-

decision bits. Alternatively, one can represent them with soft decision bits, and include the values of these bits inside the states of the trellis-decoding algorithm. This leads to a complexity increase to the exponential number of states as a function of the number of bits included in the trellis state.

Also, fewer iterations can be done, e.g. one iteration, during the 2nd, 3rd,.., Turbo equalization loop. In the case of Turbo iteration described in the cited reference, it is not necessary to do the same number of iterations during each Turbo loop. It was observed that during later iterations of the Turbo loop, the number of iterations of the bit-detector could be reduced from three to one.

Providing side information bit estimates from some other preferably low complexity detector can advantageously accelerate the start the first iteration. In the implementation of the stripe wise bit-detector illustrated in Fig. 5, during the first iteration, all unknown tracks that are used as side information are arbitrarily set to zeroes values. The reason, why during successive iterations the reliability of the bit decisions improves, is that the quality of the side information input during that iteration has improved. Thus, not using the arbitrary all zeroes initialization of the unknown bits, but use some (simple) estimates of these bits, e.g. based on threshold decisions (as addressed before) can quick start the algorithm.

Alternatively, during the first iteration, the side information bits (at one edge) of the tracks can be let uncertain by giving these bits the special value 'unknown' instead of '0' or '1'. Then, the channel output value given one or more unknown input values within the range of the channel response function can e.g. be averaged over all values of the unknown bits. E.g. more likely values ('0' or '1') can get a larger weighting in this average. In fact, this weighing may depend on soft information output by previous iterations of the bit-detector.

In another embodiment, the number of iterations can be made dependent on the estimated reliability of the received or of the estimated data. Depending on the characteristics of the channel response, it may happen that the number of iterations that is required to reach an acceptable BER performance, can be reduced if the signal to noise ratio (SNR) is high. Alternatively, one can observe the number of bits that changes value from iteration to iteration and decide to stop the algorithm if the number of bit-changes drops below a certain threshold.

Instead of shifting from top track to bottom track, one can alternatively shift from outer tracks to inner tracks. The use of reliable side information, such as a guard space, significantly improves the reliability of bit estimates and therefore is a good starting point for the iterations. Given that a section of the big spiral has guard space

at both radial ends, it is in general better to process the tracks starting at the outer tracks adjacent to the guard space and proceed to the middle track. But in special circumstances, it may be advantageous to start iterating close to a track that can be estimated with high reliability for reasons that are independent with the guard space. When, e.g. the middle of the e.g. 10 tracks in a big spiral is coded differently, e.g. with a higher degree of redundancy, or with a lower density of the pits/lands, such a track can be decoded with a higher reliability. Then, such a track can serve as a starting point for the bit-detection iterations. In such case, the shifting of the stripe of the stripe-wise detection can be done starting from two more reliable bit-rows (e.g. the guard band row, which is absolutely reliable, and the middle "more reliable" row in the example above) where the stripes move towards each other.

It is also possible to disperse the "more reliable" bits over the entire broad spiral pseudo-randomly at known locations in the format (and not just in a single row as in the example above), where these "more reliable" bits act as pinning centers for the reliability of the stripe-wise bit-detection.

A very simple generalization of the above is to make the stripes more than two tracks high, e.g. three tracks, and shift the trellis-based bit detection machine down (up) by more than one track at a time.

It is also possible to use other tracks than the top track or bottom track as output track of a stripe, as for example, to use stripes of three tracks, and use (only) the middle track of the obtained bit-estimates from the stripe detector, instead of the track that is closest to the guard space (or other reliable tracks).

The tracks can also be processed in different order during successive iterations. In the embodiments described above, the same decoding procedure was simply reiterated three times. Various alternatives can be performed such as varying the bit-detection procedure from iteration to iteration.

The invention also applies to multi-level recording. It is not essential for the principle of the invention, that the recorded information, denoted symbols, on the disc is binary ("pit"/"land"). Also non-binary, i.e. multi-level detection can be performed using the stripe wise approach. In this case, a symbol detector is used instead of a bit-detector. The cardinality of the state space in the trellis will increase exponentially in the cardinality of the alphabet of the letters for the non-binary case. When in the stripe wise bit detector, Q-ary symbols are used instead of binary symbols, notably the integers 0,1,2,....,Q-1, the number of (state, branch) pairs to be dealt with in one stage of the trellis would grow as a power of Q. E.g. if there are 4 symbols in the state and 2 symbols in the branch, the mentioned number would grow as $Q^{(4+2)}=Q^6$The stripe wise

approach is also suited when the message bits have been integrated modulo-2 ("differentially encoded") before they were recorded on the disc. The stripe wise decoder can then directly decode the message bits, i.e. do the differentiation. When one message bit is estimated erroneously in a side information track, this causes all subsequent channel input bits in that side information track to invert. When the side information bits have an influence on the channel outputs within the stripe, such an inversion will cause the distance between the corresponding hypothetical channel outputs in the stripe and the real channel outputs (i.e. the metrics) to increase. Thus, pressure is exerted by the trellis-decoding algorithm on such inversions in order not to occur, or be of short length.

The invention also applies to ISI impulse responses (both of the linear type and of the non-linear type) that not only use nearest neighbors cells, but also second nearest neighbor cells. In the embodiments described previously, side information of one track was used at either side of the stripe, because it was assumed that the hypothetical channel outputs in a stripe could be adequately approximated as functions of the channel input bits in the stripe and in the side information tracks. That is, a channel output at a certain point on the disc can be approximated as a function of the channel input bit (cell closest to the output sample), and the bit values of the nearest neighbors of that cell. If the channel response (ISI) is a more smeared out function, with a larger support, it will be preferable to use more than one side information track at either side of the stripe.

The invention can also be applied when a pre-equalizer filter ("Partial Response" equalization) precedes the bit-detector. In this case, for the computation of the hypothetical channel output values, knowledge of the channel output function is needed, or an approximation thereof, for the input bits that have a significant influence on the channel output values. The latter number may be reduced by using a linear or non-linear channel post-filter which concentrates the dependency of the channel output plus post-filter in a small number of channel inputs positions. In the literature, such filtering techniques are well known and are called partial response techniques.

The bit-detection process that is carried out within a stripe with the mentioned trellis-based decoding algorithms is not described in detail, since it is well known. It is described for example in [1] . A Viterbi-decoding algorithm is also described in E.A. Lee and D.G. Messerschmit "Digital Communication" 2nd Edition Kluwer Academic Publishers 1994 Boston/Dordrecht/London Section 9.6. [3]. The article by G.D. Forney, Jr., "Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference", IEEE Trans. Inform. Theory, Vol. IT-18, No. 3, pp. 363-378,

May 1972 [4] also describes trellis-based decoding algorithms. It is recalled that Gaussian channel output noise leads to Mean Square Estimates being produced by some of the mentioned algorithms (Viterbi, (Max-)(Log-)MAP). In case of non-Gaussian channel output noise, the trellis decoding algorithms mentioned can still be applied, but the metric functions must be adapted to the probability density function of the noise.

Finally, for a stationary channel, the hypothetical channel output values as a function of the previous trellis state, the next trellis state (or the branch value), and bit values in the side information tracks within the significant input bit range of the channel outputs in the stripe can be tabulated to prevent re-computation of these channel output. The algorithm is also applicable to non-stationary channels. Then, such a table needs to be updated when the channel output function has changed significantly.

Also, sub-optimal per-stripe bit-detection algorithm could be used instead of Viterbi, (Max-)(Log-)MAP or SOVA., e.g. based on the M-algorithm or on the T-algorithm (to process only a subset of all states in a given section of a trellis), or on the Fano or the stack algorithm.

Fig. 7 shows curves of the Bit Error Rate (BER) versus Peak Signal to Noise Ratio (PSNR) performance of various bit-detectors. The left-most (blue) curve shows the performance of the stripe wise detector with stripes of two tracks, and three iterations during the first Turbo loop and 1 iteration during successive Turbo loops with a rate 0.90 low density parity check (LDPC) code. The second-to-left (green) curve shows the performance of the same stripe wise bit-detector without Turbo equalization. The other curves show the performance of the best-known point-wise detectors. The red curve shows the performance with a rate 0.90 LDPC code. The right-most black curve shows the performance without error correcting codes. Finally, Fig. 7 shows that the stripe wise detector in accordance with the invention has a performance superior to that of all other detectors known so far.

The present invention enables achieving reliable symbol detection by iterating a stripe wise (multi-row) trellis-based symbol detection method, one iteration representing an application of the trellis-based symbol detection method along a stripe of a sub-set of rows in the channel tube. Interference between successive neighboring symbol rows is taken into account as side information in the computation of the branch metrics of the trellis (for the considered symbol row).

The drawings and their description herein before illustrate rather than limit the invention. It will be evident that there are numerous alternatives, which fall within the scope of the appended claims. In this respect, the following closing remarks are made.

There are numerous ways of implementing functions by means of items of hardware or software, or both. In this respect, the drawings are very diagrammatic, each representing only one possible embodiment of the invention. Thus, although a drawing shows different functions as different blocks, this by no means excludes that a single item of hardware or software carries out several functions. Nor does it exclude that an assembly of items of hardware or software, or both carries out a function.

Claims.

1. A symbol detection method for detecting the symbol values of a channel data stream recorded along an N-dimensional channel tube, N being at least 2, on a record carrier of a set of symbol rows, one-dimensionally evolving along a first direction and being aligned with each other along at least a second of N-1 other directions, said first direction together with said N-1 other directions constituting an N-dimensional lattice of symbol positions, the method comprising iterative stripe-by-stripe application of a trellis-based symbol detection step, wherein a stripe is a subset of at least one neighboring row.

2. A method as claimed in claim 1, wherein N is 2 and the symbols of said channel data stream are arranged on a two-dimensional hexagonal or square lattice.

3. A method as claimed in claim 1, wherein said trellis-based symbol detection step includes application of Viterbi, (Max-)(Log-)MAP, SOVA algorithms and their derivatives and simplifications such as the M- and T- or stack-algorithm.

4. A method as claimed in claim 1, wherein an iteration of said stripe wise trellis-based symbol detection step comprises:

- estimating symbol values in a first subset of rows, denoted current subset, using a trellis-based algorithm, side information derived from at least one row adjacent to said current subset being used in the estimation of said symbol values,
- shifting to a second subset of rows.

5. A method as claimed in claim 4, wherein said second subset of rows has at least one row in common with the first subset of rows.

6. A method as claimed in claim 4, wherein estimated symbol values of the current subset of rows obtained from the current iteration serve as side information in the estimation of the symbol values for the subsequent iteration on the subsequent subset of rows.

7. A method as claimed in claim 4, comprising meta-iterations of said iterative stripe wise trellis-based symbol detection step using estimated symbol values obtained from a previous meta-iteration as side information in the estimation the symbol values for the

current meta-iteration, the number of rows in the subsequently processed subsets of rows and/or the order within which the subsets are processed from one iteration to another one being defined for each meta-iteration.

8. A method as claimed in claim 4, wherein said symbol values are recorded along a two-dimensional channel tube forming a multi-row wide spiral comprising a set of adjacent symbol rows aligned with respect to each other in the radial direction, a symbol-free guard space row being provided between successive revolutions of the multi-row wide spiral, and wherein said side information is derived from said guard space row.

9. A method as claimed in claim 8, wherein the side information derived from said guard space row is a predefined fixed value.

10. A method as claimed in claim 8, wherein said first subset of rows is adjacent to the guard space row.

11. A method of reproduction of a user data stream, which is recorded on a record carrier, comprising a symbol detection method as claimed in claim 1 for detecting the symbol values of bits of said channel data stream.

12. A symbol detector for detecting the symbol values of a channel data stream recorded along an N-dimensional channel tube, N being at least 2, on a record carrier of a set of symbol rows, one-dimensionally evolving along a first direction and being aligned with each other along at least a second of N-1 other directions, said first direction together with said N-1 other directions constituting an N-dimensional lattice of symbol positions, the symbol detector comprising means for estimating symbol values in a first subset of rows, denoted current subset, using a trellis algorithm, side information derived from at least one row adjacent to said current subset being used in the estimation of said symbol values.

13. A reproduction device for reproduction of a user data stream, which is recorded on a record carrier, comprising a symbol detector as claimed in claim 12 for detecting the symbol values of bits of said channel data stream.

14. A receiver for receiving symbols from a channel data stream, the receiver comprising a symbol detector as claimed in claim 12 for detecting said symbol values received from said channel data stream.

15. A recording medium for use in a receiver as claimed in claim 14, for storing symbol values received from a channel data stream.

16. A system comprising a source for delivering symbols through a channel data stream and a receiver for receiving said symbols from said channel data stream, wherein said receiver comprises a symbol detector as claimed in claim 12 for detecting the symbol values received from said channel data stream.

17. A computer program product for a receiver computing a set of instructions which when loaded into the receiver causes the receiver to carry out the methods as claimed in claim 1 and 11.

18. A signal for carrying a computer program, the computer program being arranged to carry out the methods as claimed in claims 1 and 11, when said computer program is executed on a computer.

19. Making available for downloading the computer program as claimed in claim 17.

Abstract.

The invention relates to trellis-based symbol detection for detecting symbols of a channel data stream recorded on a record carrier. Typically, it applies when symbol values are recorded along a two-dimensional (2D) channel tube forming a multi-track wide spiral comprising a set of adjacent symbol tracks aligned with respect to each other in the radial direction. The invention enables to do a kind of per "small number of" tracks equalization, so-called wise equalization to decode the 2D information encoded on the broad spiral.

Fig. 6

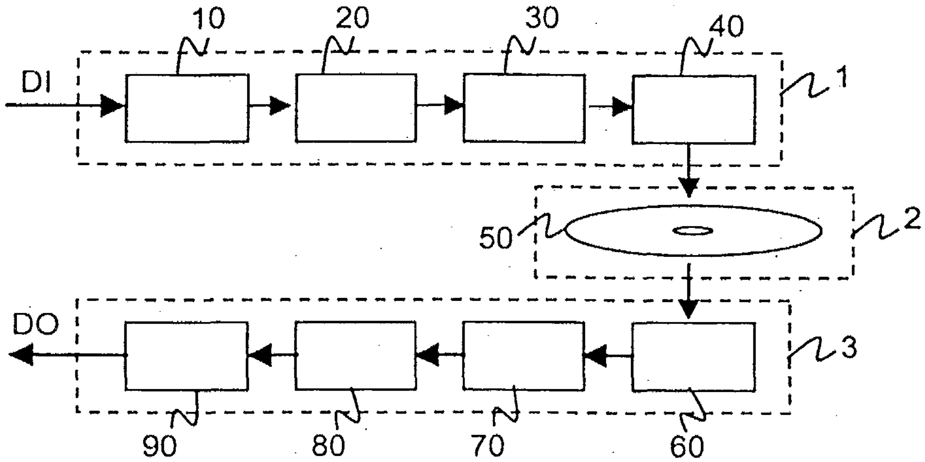


FIG. 1

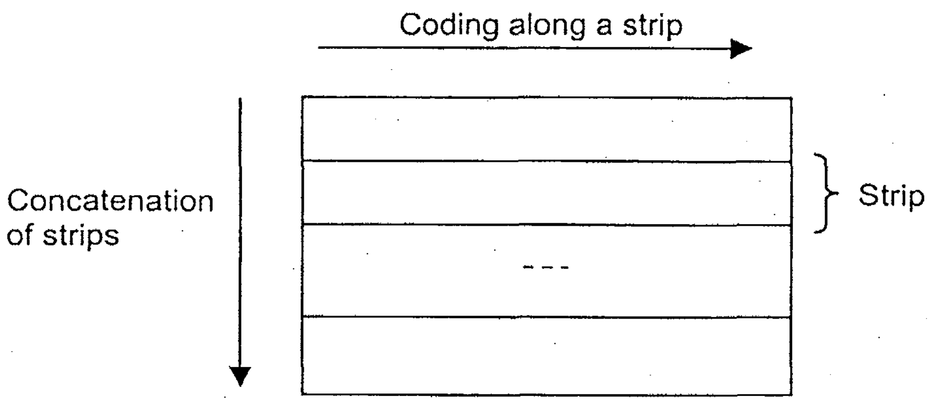


FIG. 2

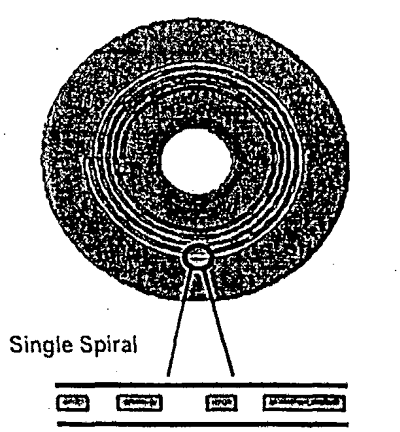


FIG. 3A

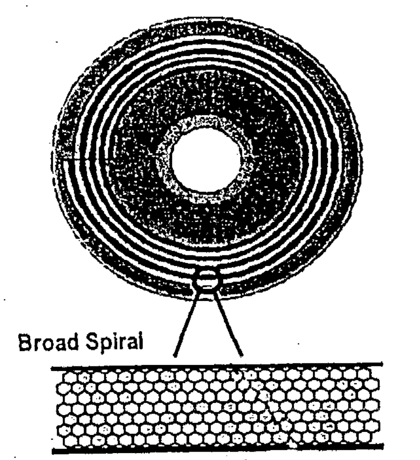


FIG. 3B

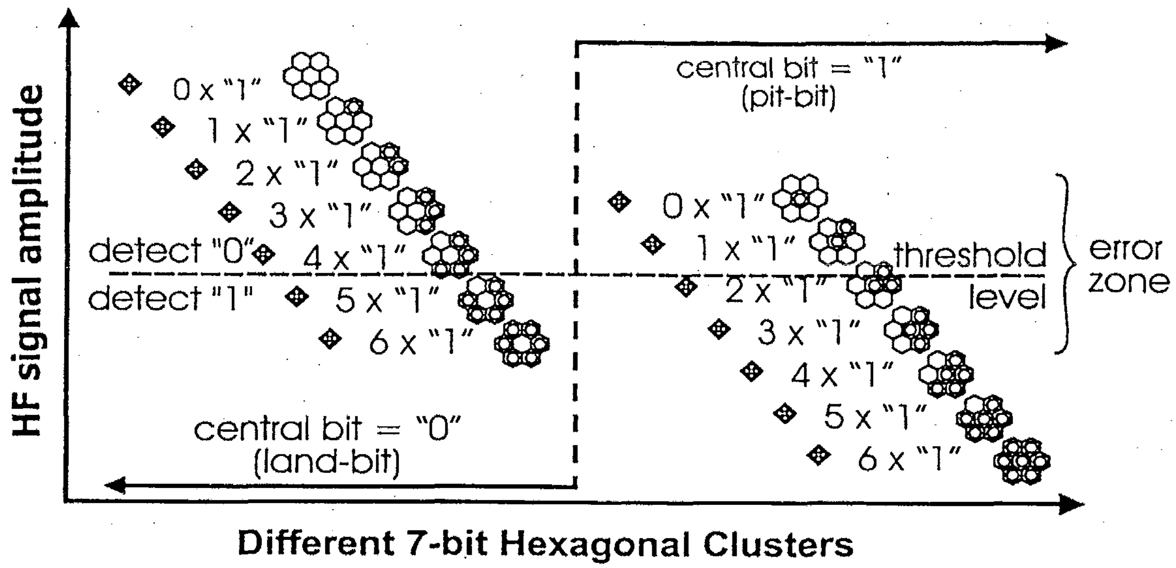


FIG. 4

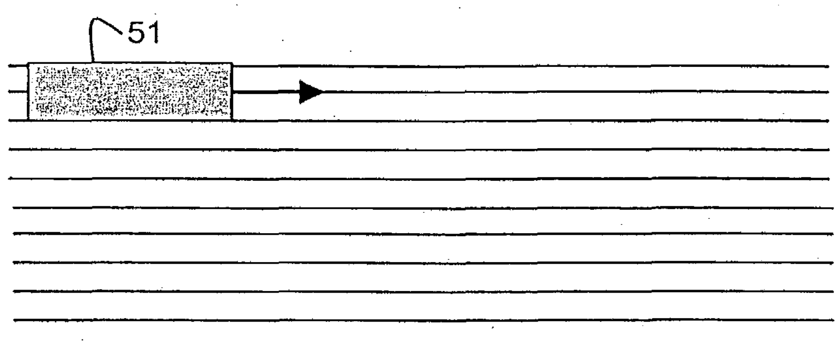


FIG. 5

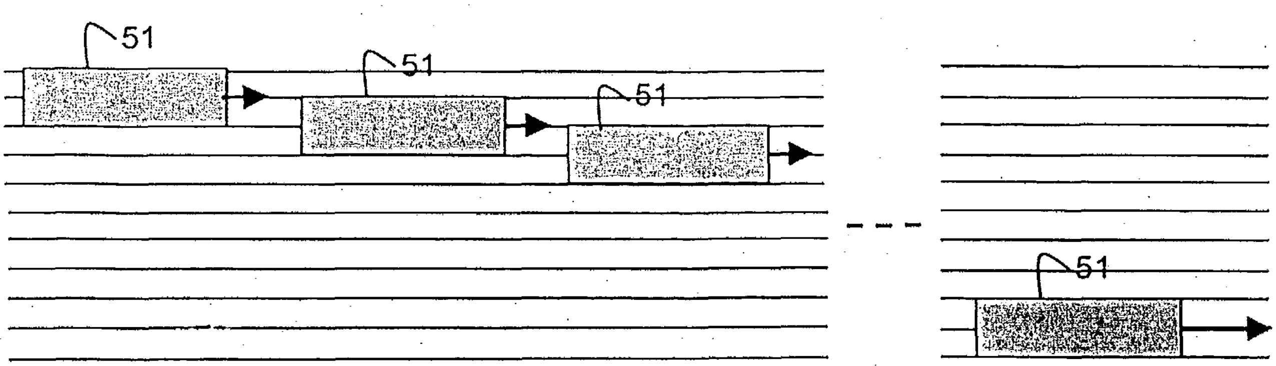
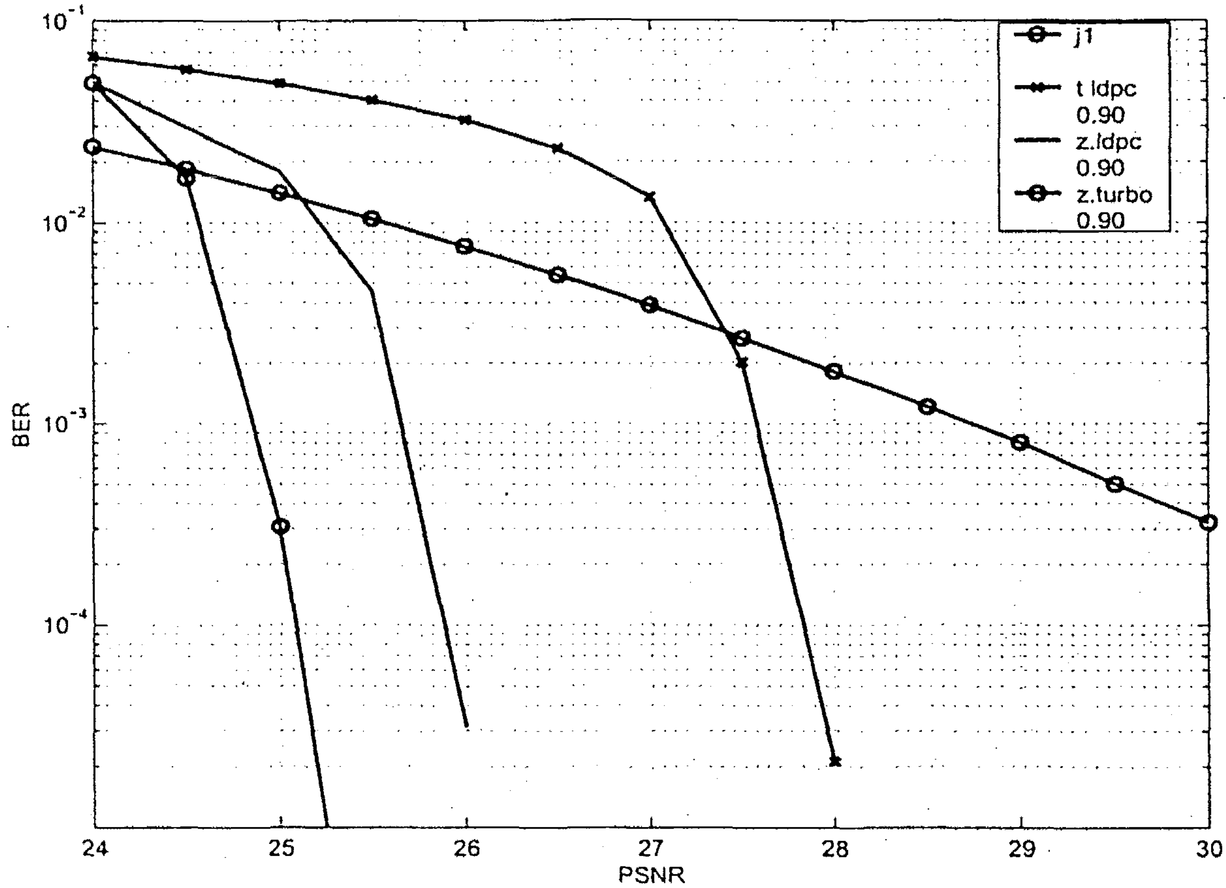


FIG. 6



10^-1
10^-2
10^-3
10^-4
BER
24
25
26
27
28
29
30
PSNR
j1
t ldpc 0.90
z.ldpc 0.90
z.turbo 0.90

FIG. 7